SENSOR FUSION IN TELEROBOTIC TASK CONTROLS

Antal K. Bejczy and Won S. Kim
Jet Propulsion Laboratory
California Institutc of Technology
Pasadena, CA 91109

Abstract

This paper describes a display and control methodology for combining (or "fusing") different multidimensional sensor data to guide the performance of telerobotic contact or near-contact tasks successfully in both manual and supervised automatic modes of control. Success is defined as a mapping of control goals or subgoals into a multidimensional data space. Several implemented examples arc presented and illustrated. The methodology can also be extended to virtual reality simulation of telerobotic task scenarios through proper physical modeling of sensors.

1. INTRODUCTION

Model-basccl tclcrobotic task control and automation is typically limitcd to (i) motion control of robot arms in free space and (ii) contact events under very well known and predictable conditions. The majority of tclerobotic *contact* or *near-contact* task controls have to rcly upon the resolution of geometric and dynamic uncertainties *during* task execution through the usc of *different* sensor data, This leads to the problem of on-line or real-time sensor data fusion in tclcrobotic task cent]-o]s. This problem originates from the fact that the sensor data typically are *multidimensional in the* work space (for instance, X, Y, Z distances) and the workspace itself can have *different physical units or dimensions* (for instance, distance, direction, force, ctc.). On the other hand, the event of success of a typical tclcrobotic contact or near-contact task control is cxprcsscd as some *combination* or *fusion* of different sensed data,

The purpose of this paper is to present a methodology for handling some sensor data fusion in tclerobotic task controls. First, the concept of "sensor information events" will bc discussed in terms of task or subtask goals. Then the concept of event-driven displays and control algorithms will bc introduced, followed by a fcw implementation examples. The implementation examples also cover sensor data fusion in virtual (computer graphics simulated) environment.

2. SENSOR INFORMATION EVENTS

Proximity, force-torque and touch sensor data arc inherently multi-dimensional. A six-dimensional forcc-torque sensor outputs the time trajectories of three orthogonal force and thı cc orthogonal torque components normally referenced to a hand coordinate frame. The hand coordinate frame itsclf is a variable (i. e., has time trajcctories) relative to a fixcd "base" rcfcrencc frame. A multipoint proportional touch sensor measures the area distribution and amount of contact pressure over a fixed surface. A single proximity sensor measures short (few centimeters) distances in a given direction relative to a hand coordinatc frame. Several proximity sensors in a given emplacement geometı y on the hand can measure several or all six position and orientation variables of the hand relative to objects.

A sensor-rcfcrenccd or sensor-guided manipulator control task contains a goal or a set of subgoals. The control goal or subgoals arc cxprcssed as a combination of various sensor data. The simultaneous occurrence of time trajectoı ics of various sensor data at a single point or within a given subvolume of a multidimensional data space can bc called a sensoı information "event". Hence, sensor information "event" is the projection or mapping of the control goal or subgoals into a multidimensional data space.

Figure 1 gives simple illustrations for the concept of sensor information "event". Equal length of two proximity sensor beams can bc an "event' in the sense that it may signify, e.g., the roll, yaw or pitch alignment of a mechanical hand relative to an object. Equal magnitude of two orthogonal force components can bc an "event" in the sense that it may signify, e.g., the push or pull of an object by a mechanical hand in a given direction. Or, for instance, half contact coverage of a touch-sensitive area on a mechanical finger can bc an "event" in the sense that it may signify, e.g., that there is sufficient contact between hand and object for successful grasp. Typically when an "event" occurs, some control action must bc taken.

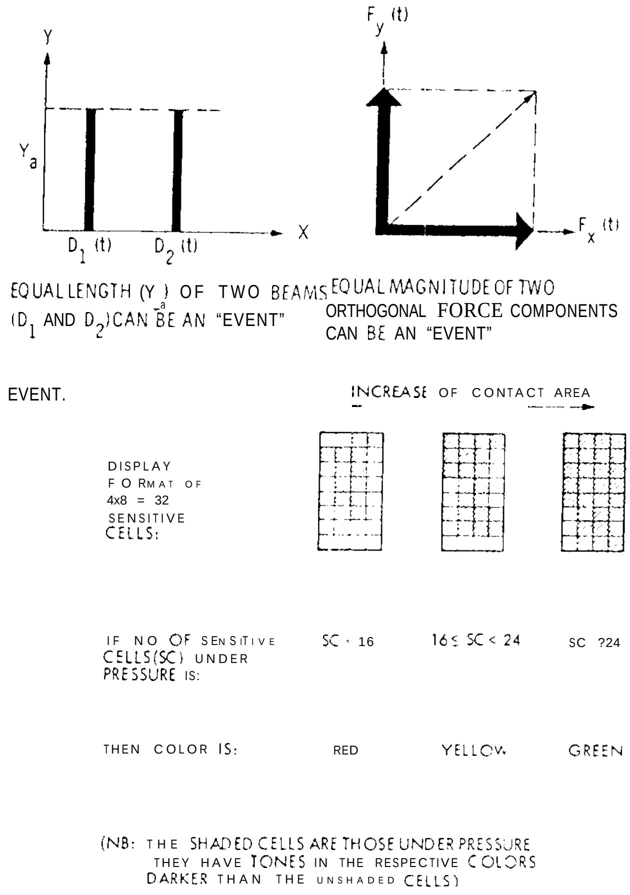

Y
Y_a
D_1 (t)
D_2 (t)
X
F_y (t)
F_x (t)
EQUAL LENGTH (Y_a) OF TWO BEAMS (D_1 AND D_2) CAN BE AN "EVENT"
EQUAL MAGNITUDE OF TWO ORTHOGONAL FORCE COMPONENTS CAN BE AN "EVENT"
EVENT.
INCREASE OF CONTACT AREA
DISPLAY FORMAT OF 4x8 = 32 SENSITIVE CELLS:
IF NO OF SENSITIVE CELLS(SC) UNDER PRESSURE IS:
SC · 16
16≤ SC < 24
SC ?24
THEN COLOR IS:
RED
YELLOW
GREEN
(NB: THE SHADED CELLS ARE THOSE UNDER PRESSURE THEY HAVE TONES IN THE RESPECTIVE COLORS DARKER THAN THE UNSHADED CELLS)

Figure 1. Simple Examples of Sensing-Defined "Events"

From the viewpoint of telerobotic task control, sensor information events pose two different problems in automatic and in manual modes of operation. In automatic control, the problem is to formulate and implement a control algorithm that uniquely drives the systcm to the desired goal as expressed by a sensor information event. Taking a desired control action once the "event" occurs is a simple task in automatic control since the event automatically will trigger the required control action (e.g., "stop motion"). To the contrary in manual control, the problem is to *take* the control action once the "event" occurs since, to do so, the operator has to follow and evaluate a multidimensional set of data in real time, This sensor data fusion, which creates the desired "event" information, is a demanding task and heavy workload for the operator, and a common source of control errors. The problem is now to implement sensor driven event displays for the operator in easily perceivable and unmistakably unique formats.

3. SENSOR-I) RIVI:N EVENT DISPLAYS

Event-driven displays can be implemented by developing and/or employing appropriate real-time algorithms which (a) coordinate and evaluate sensor data in terms of predefined "events" and, (b) drive some appropriate information display ili-real time. Manipulator control tasks can be subdivided into a multitude of sensor "events", and each event may have a variety of characteristic parameters. Thus, the development of fairly general purpose event-driven displays requires that the logic/parametric structure of the algorithms be flexible in the sense that changing control goals or subgoals can be accommodated by simple call-changes in the algorithms in a given control/operation environment.

The actual event display can be implemented by alternative means, the selection of which depends on the application environment. For event displays, both audio and visual display techniques arc suitable. An important consideration for selecting and designing event displays is the "warning effect" the display shall impose on the operator. By definition, the occurrence of a sensor event should call the operator's attention to some appropriate control decision and action, without disturbing the operator's normal visual attention directed toward the overall control task. Note that the control can require split-second decisions. Another important consideration is related to the selection of the *content* of the display format. How much and what kind of *detailed* information the operator should be exposed to in addition to the "event information" within the same general frame of information? Too much information can be disturbing. Too little information can defy the purpose. The display of uncorrelated data, or the display of correlated data in uncorrelated form, may impose heavy cognitive load on the operator.

Properly designed event-driven displays arc expected to have a number of benefits: (a) simplify on-line control decisions; (b) reduce errors caused by human factors; (c) reduce pcrceptua]/cognitive workload on human operator in a real-time control environment; (d) improve overall control performance in control situations which many times require split-second type control decisions.

It is noted that the algorithms that operate the sensor driven event displays arc intimately related to the control algorithms when the operation is in automatic mode of control since the same algorithms that show the event to the operator will also trigger the event-related action in automatic mode of control. Moreover, when the displays also show the event-referenced error together with the corrective action needed to eliminate the error then, in supervised automation, the operator can efficiently exercise the supervisory function by viewing the sensor driven event displays.

4. IMPLEMENTATION EXAMPLES

Several sensor driven event displays have been implemented in the Advanced Teleoperator project at the Jet Propulsion Laboratory (JPL) in the past, for both manual and automatic modes of control operation. A few examples arc quoted below.

4.1 Event-Driven Proximity Displays

~vcnl-driven displays were implemented for two types of four-sensor emplacement configurations. Figure 2 shows, for the *first type of emplacement geometry,* the general format of graphics display of four proximity sensor data. The display shows the "bone" of a parallel jaw hand and four beams emanating from the hand, two orthogonal beams from the tip of each jaw. The beam lengths arc proportional to the sensitive length of the sensor beams. Each beam length is bound to about 10 cm distance. Figure 3 summarizes the proximity events together with the event logic and event parameters that have been implemented. The parameter D is fixed at 5 cm. D is always defined parallel to and halfway in between the two beams which measure roll and yaw alignments, respectively, and relative to the line connecting the two fingertips. The tolerance, T, can be set by inputs on the computer. Values from 0.5 to 7.5 cm arc allowed. Any combination of the four event logic equations may be selected to control the event success blinker. The success may be defined as X alignment with a tolerance, say, of 1 cm (corresponding to about 5 degrees when the hand is fully open). Or, the success may be defined as Y range of 5 cm together with X alignment to within 0.5 cm tolerance (corresponding to about 2.5 degrees when the hand is fully is fully open). This latter "success case" would be useful in moving the hand over a table to a wall while holding an object vertical. With this event logic, the hand roll angle would be small as the range measurement is made on both sensors and the object would be held with the hand 5 cm above the table. The final approach to the wall would bc reached with the hand perpendicular to the wall.

Figure 4 shows two uscs of the event-driven proximity display. The first photograph (Figure 4A) shows the hand above a table and skewed to a block. The task is to achieve alignment with the table and the block. The second photograph (Figure 4B) shows that this has occurred and the event blinker has come on. The third photograph (Figure 4C) shows a different alignment problem, Here, it is desired to bring the hand in level over the plate on the table. There are no forward references. When the desired level state is achieved, the event blinker comes on as shown in the fourth photograph (Figure 41)).

The information aspects of object encounter phase motion, using four proximity sensors, arc shown in Figure 5. Here a manipulator is approaching a block resting on a table, and the operator must align the hand to the block prior to grasping it. Without the ability to randomly position TV cameras, the operator will I need precise information about the relative location of the hand to the table, and the relative location and orientation of the hand to the block. The dashed lines in Fig. 5 indicate the lines of sight of four finger mounted proximity sensors. The coordinates associated with the encounter regime arc also shown in this Figure. The di's arc the path lengths detected by the proximity sensors. As

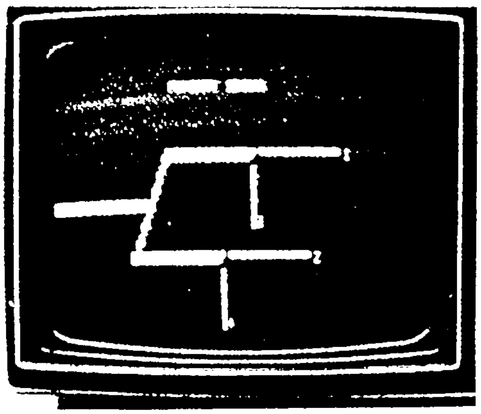
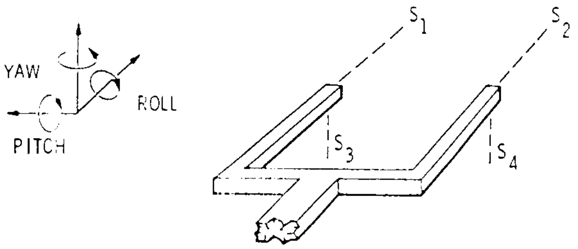


Figure 2. Display of Four Proximity Sensor Beams on a Parallel Jaw Hand



S_i = RANGE MEASURED BY SENSOR "i", i = 1, 2, ? 4

D = DISTANCE } PRESET PARAMETERS
T = TOLERANCE }

LOGIC DEFINITIONS FOR EVENT E_i = {K}

E_i . 0 FOR $K \geq 0$: EVENT IS N'T THERE, BLINKER 'OFF

E_i = 1 FOR $K < 0$ EVENT OCCURED, BLINKER "ON"

EVENTS	LOGIC EQUATIONS
1. YAW ALIGNMENT AT "D" WITH "T" TOLERANCE	$E_1 = \{ \lvert S_1 - D \rvert - T > 0 \} \bullet \{ \lvert S_2 - D \rvert - T > 0 \}$
2. YAW ALIGNMENT ONLY WITH "T" TOLERANCE	$E_2 = \{ \lvert S_1 - S_2 \rvert - T > 0 \}$
3 ROLL ALIGNMENT AT "D" . ITH "T" TOLERANCE	$E_3 = \{ \lvert S_3 - D \rvert - T > 0 \} \bullet \{ \lvert S_4 - D \rvert - T > 0 \}$
4 ROLL ALIGNMENT ONLY WITH "T" TOLERANCE	$E_4 = \{ \lvert S_3 - S_4 \rvert - T > 0 \}$

Figure 3. Definition of Some Proximity Sensing Events

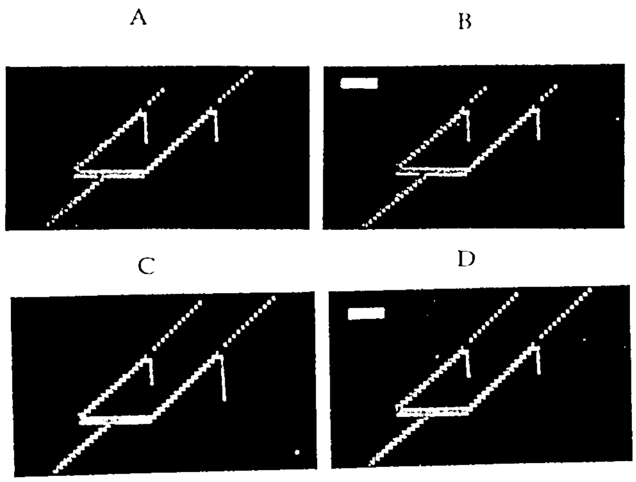


Figure 4. Uses of Event Driven Proximity Displays

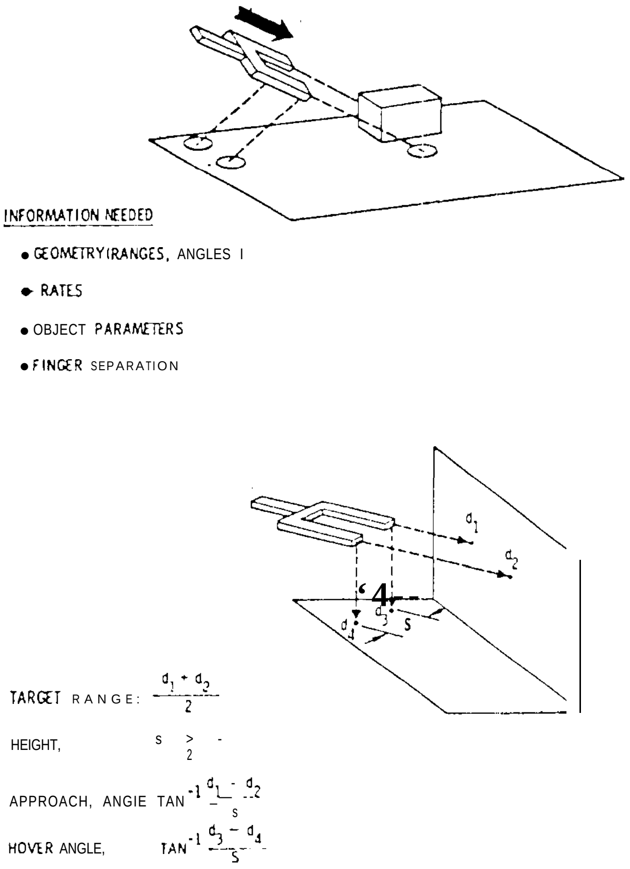

INFORMATION NEEDED
● GEOMETRY(RANGES, ANGLES I
● RATES
● OBJECT PARAMETERS
● FINGER SEPARATION
d_1
d_2
d_3
d_4
S
TARGET RANGE: $\frac{d_1 + d_2}{2}$
HEIGHT, s > - 2
APPROACH, ANGIE TAN^{-1} $\frac{d_1 - d_2}{s}$
HOVER ANGLE, TAN^{-1} $\frac{d_3 - d_4}{S}$

Figure 5. Information Aspects of Encounter Phase Motion Using Proximity Sensors

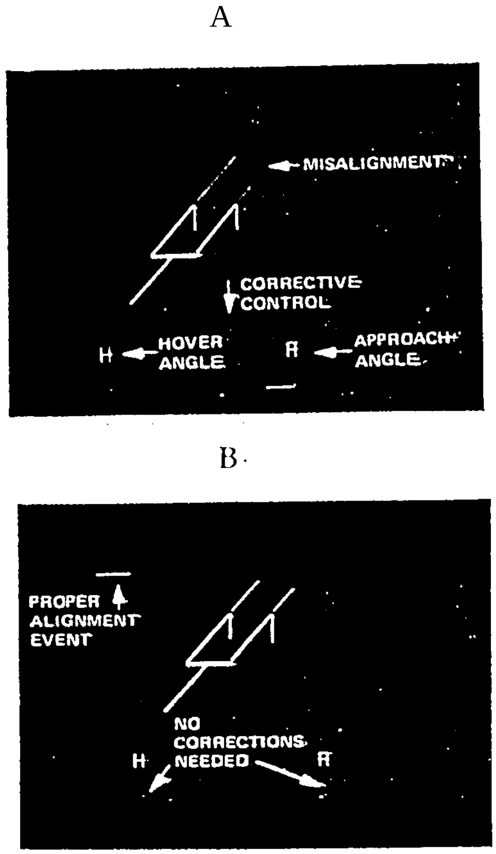


Figure 6. Encounter Proximity Display with Indication of Corrective Control

seen from the wrist of the manipulator the approach angle is equivalent to yaw and the hover angle to roll. By changing the separation bctwccn the fingers, the d_1 and d_2 measurements can be used both to find the approach angle and find the sides of the object (block).

Figure 6 shows the encounter proximity control display, also indicating the corrective action needed for alignment. The hand is shown schematically together with four bars indicating the d_i's of Figure 5. The bar lengths arc proportional to tbc d_i's. At the bottom of the display the required corrective control is shown. In Figure 6A a large approach angle (yaw) error is shown, In Figure 6B that error has been eliminated. The error is much easier to sec from the automatically monitored error bar than it is from comparing the relative lengths of the d_i's visually.

Figure 7 shows the *second type of emplacement geometry* of four proximity sensors in a plane and the definitions of depth, pitch and yaw errors related to this square-symmetric sensor emplacement geometry. Note that all errors arc referenced to a flat plane. Suppose that a

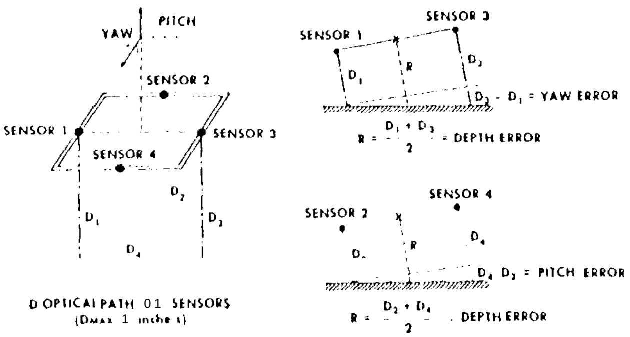


Figure 7. Four-Sensor Operation Concept for Simultaneous Measurement of Depth, Pitch and Yaw Errors

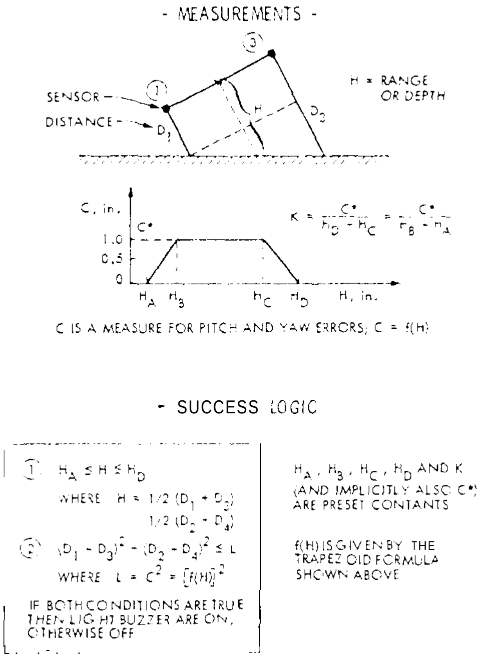

- MEASUREMENTS -
SENSOR
DISTANCE
H = RANGE OR DEPTH
D1
D3
H
C, in.
C*
1.0
0.5
0
HA
HB
HC
HD
H, in.
K = C*/(HD - HC) = C*/(HB - HA)
C IS A MEASURE FOR PITCH AND YAW ERRORS; C = f(H)
- SUCCESS LOGIC
1 HA ≤ H ≤ HD
WHERE H = 1/2 (D1 + D3)
1/2 (D2 + D4)
2 (D1 - D3)² - (D2 - D4)² ≤ L
WHERE L = C² = [f(H)]²
IF BOTH CONDITIONS ARE TRUE THEN LIGHT BUZZER ARE ON, OTHERWISE OFF
HA, HB, HC, HD AND K (AND IMPLICITLY ALSO C*) ARE PRESET CONTANTS
f(H) IS GIVEN BY THE TRAPEZOID FORMULA SHOWN ABOVE

Figure 8. Conic Algorithm for indicating Acceptable Combinations of Range, Pitch and Yaw Errors for Successful Grasp for Sensor Configuration Shown in Fig. 7.

successful grasp permits given ± pitch and yaw errors and a given H distance error for a given end effector and a given grapple fixture. Several "success algorithms" can be defined for this case. For the sake of brevity, only one "success algorithm" is shown in this paper, summarized in Figure 8. It is called the "conic algorithm" since it condenses the individually allowable pitch and yaw errors into a simple allowable cone angle error condition. (See Condition 2 in Figure 8.) Three kinds of "success definitions" have been developed, each with three sets of "success parameters". All

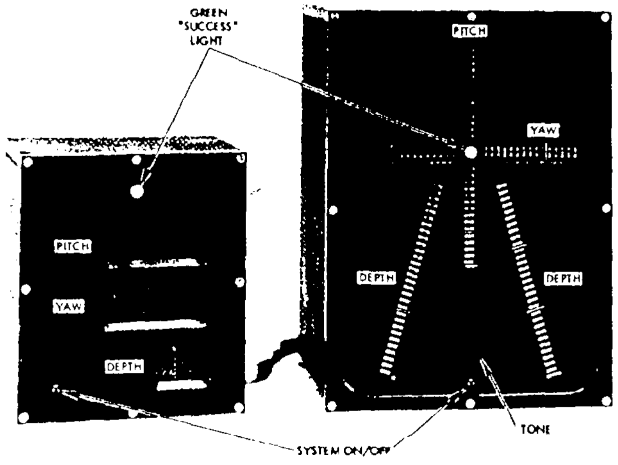


Figure 9. Event Display Format for Sensor Configuration Shown in F igure 7

nine variations have been implemented for "all four" and for "three-out-of-four" sensors. All together 18 algorithms arc stored in a microcomputer. Any one of the 18 algorithms were easily callable on the keyboard, The three "success definitions" arc related to the rotational symmetry involved in the pitch and yaw definitions. The three sets of "success parameters" arc related to the three sets of permissible ± a and H errors in order to evaluate the control system's responsecharacteristics. Note also that the four-sensor square-synnnetric emplacement configuration is redundant for the definition and computation of depth, pitch and yaw *errors.* A triangular configuration of three sensors would bc sufficient for that purpose.

A simple "success display" (tone or green light) would not show the details of the three-dimensional (depth, pitch and yaw) error states. Figure 9 shows a simple display format which indicates both the error states and their "successful" combination for a selected set of parameters.

4.2 Event-lhiven Force-Torque Display

A six-dimensional force-torque sensor located at the base of a mechanical hand provides a complex information on the. mechanical hand's interaction with objects and environment. Consider the task of sliding a block in a groove across a table by pushing it. (See Figure 10.) The applied forces must bc in the direction of the groove if the block is to bc moved efficiently and safely. Figure 10 also shows an appropriate "event-driven" display. When the forces arc applied correctly, the operator will know it by the event indicator. If not, the operator will scc the force errors and bc able to apply the needed corrections.

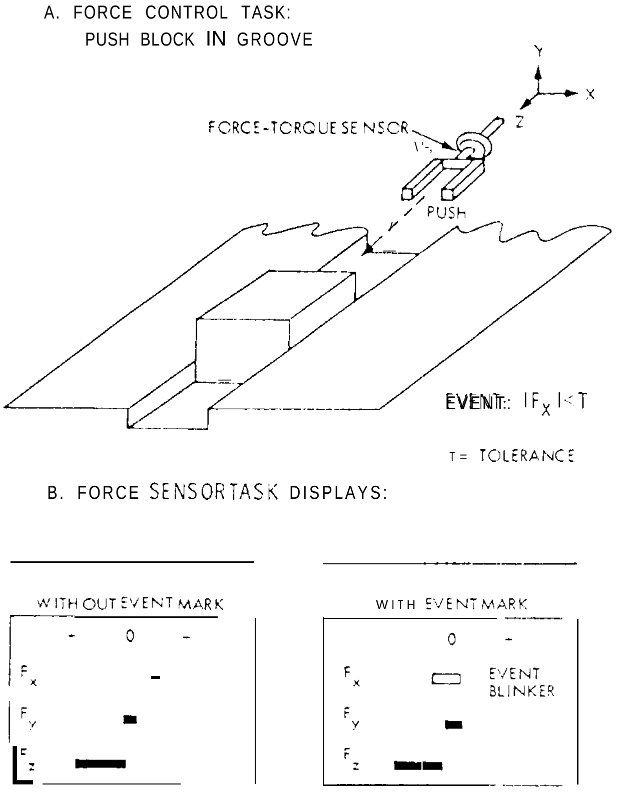


Figure IO. Force-Torque Event Display ample

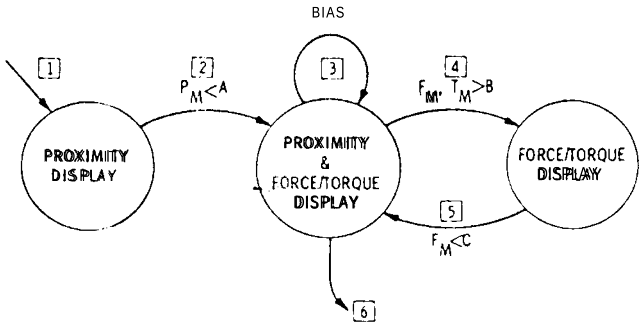


Figure 11. Event-Controlled Display Transition Logic Diagram for Object Approach, Acquisition and Removal

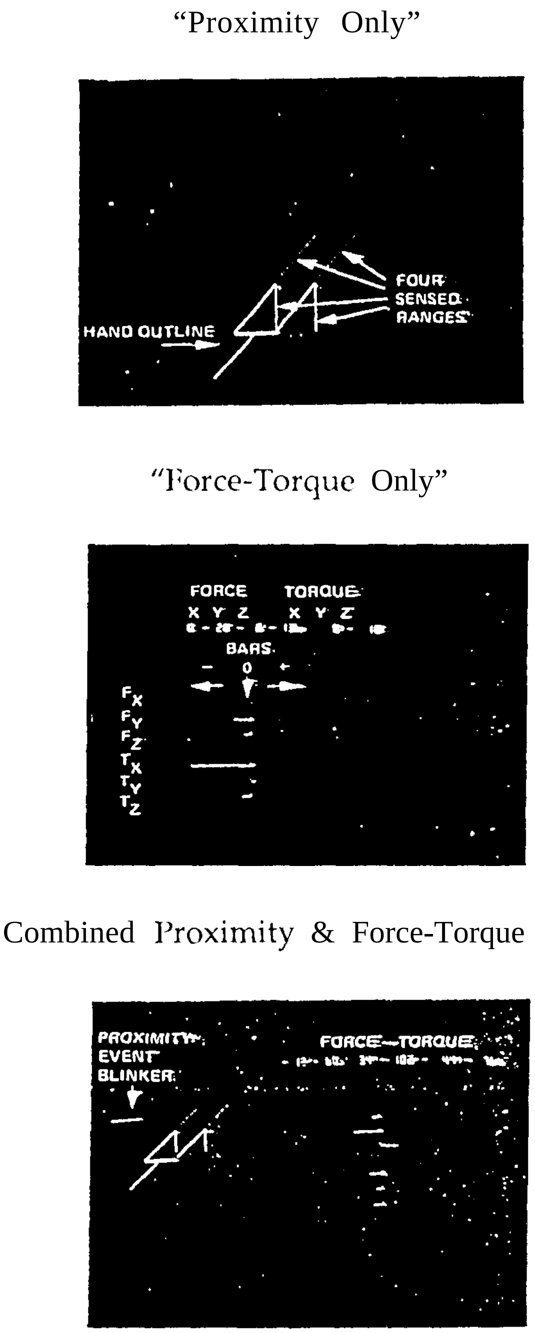
4.3 Event-Controlled Displays

Throughout a telerobotic task performance several different displays and display modes may be required. For instance, to move the robot hand within an object encounter area, proximity event displays are needed. When the object is being contacted, tactile and/or force-torque event displays are needed. Event-controlled displays can *automatically* effect changes in both the types of displays and display parameters, matching the particular information needs related to different phases of a task execution.

Figure 11 shows a transition logic diagram for effecting changes in the types of displays that are normally needed to perform a complete task which can be divided into six subtasks. At stage (1), the manipulator is in the safe condition. The proximity only display is enabled to focus the operator's attention on getting into the encounter regime and aligning the manipulator to the object without accidental collision. When alignment has occurred, stage (2), the display will show both proximity and forces and torques. Together with the TV display this allows the object to be touched without large unknown forces and while maintaining alignment. At this point, stage (3), the operator removes the biases in the force/torque readings caused by cables and gravity and initiates the grasp process. As the object is grasped, knowledge of the forces and torques becomes of paramount importance and the force/torque only display mode is entered, stage (4). As the object is lifted its contact with environment is reduced, stage (5), and once again the combined proximity and force/torque display is desired. As the object is moved into a still less cluttered environment, the hand biases may be restored as they are manipulator geometry and gravity vector dependent so that the operator has a full view of the manipulator loads. The manipulator is then brought into the safe region, stage (6). Note that if the object is to be placed in a new rich environment, steps (2) to (6) may be repeated several times. A substantial workload is placed on the operator to manage these display mode transfers.

Automatic event mode switching can alleviate much of the display control workload. The conditions for steps (2), (4) and (5) can be detected using event detection logic and the mode switches automated. This scheme has been implemented. Automatic mode switching reduces the operator's workload in both manual and supervised automation modes of control.

Figure 12 shows the three display types indicated on Figure 11. The "proximity sensor only" scene corresponds to the case where the baud is level over, say a table top and no object is in front of the fingers. The "force/torque only" display shows both the quantitative forces and torques, in ounces and inch-ounces respectively, as well as graphical bar representations of the data. For each bar, zero is in the center of the screen, positive data values generate a bar to the right, and negative values a bar to the left. In the combined, "dual", display both proximity and force/torque data are displayed, although in reduced scale.


“Proximity Only”

"Force-Torque Only”

Combined Proximity & Force-Torque

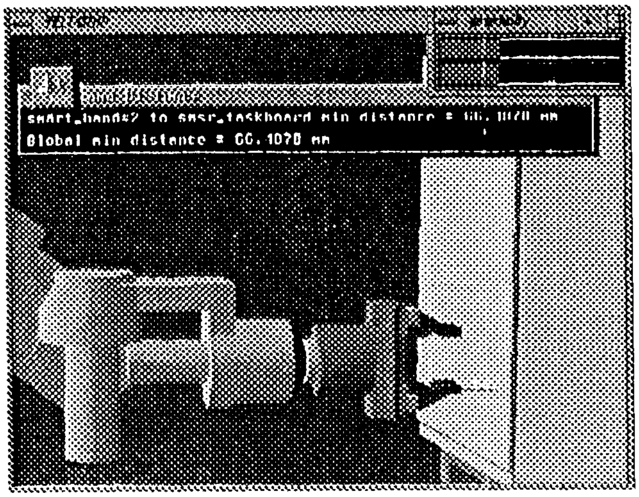
Figure 12. Three Display Types Controlled by Transition Logic Shown in Fig. 1'1 '

5. VIRTUAL REALITY SIMULATION

There is a growing interest in the application of computer graphics for telerobotic task planning and previewing and for operator training [1]. Virtual reality (that is, computer graphics) simulation of telerobotic task scenarios can save time and cost and increase confidence in preparing telerobotic operations. Using proper physical modeling, even sensors and sensor fusion can be simulated by computer graphics techniques with high fidelity.

Computer graphics simulation of proximity sensor signals is a relatively simple task since it only implies the computation of distance from a fixed point of a moving robot hand in a given (computed) direction to the nearest environment or object surface in the graphics “world model”. For this simulation, the TELEGRIP™ software package of Deneb Robotics, inc., has been adopted. The package provides an excellent interactive 3-D graphics simulation environment with CAD-model building, workcell layout, path designation, and motion simulation. It also provides various functions related to object distance computations and collision detection. TELEGRIP CLI (ascii-text Command Line Interpreter) commands include INQUIRE COLLISIONS (reporting collision and near miss status) and INQUIRE MINIMUM DISTANCE (reporting minimum distance between parts or devices). TELEGRIP GSL (Graphics Simulation language) also supports ray_ cast () function that computes the intersection distance from a point in a specified (ray cast) direction. Using these object distance computation functions, various proximity sensors can easily be simulated, as shown in Figure 13. The two black bars in the upper right corner show the calibrated distance of the two finger tips relative to the vertical surface of the block; decreased length of bars indicates decreased distance. (It is noted that the TELEGRIP™ package also contains the JPI,-developed [2] and now commercialized capability of calibrating the virtual reality graphics task models to actual TV camera views of the same task scenes.)

Force/torque sensor scan also be simulated by computing virtual contact forces and torques for given simulated geometric contact models. in general, an accurate simulation of virtual contact forces and torques can be very computfition-intensive, but an approximate simulation, for example, a simplified peg-in-hole task, can be accomplished without difficulty, as illustrated in Fig. 14, and described in detail in [1]. in this graphics simulation, the hole and its support structure are assumed to be rigid with infinite stiffness, while the robot hand holding the peg is compliant for all three Cartesian translational axes and also for all Cartesian rotational axes. (It is further assumed that the compliance center is located at a distance L from the tip of the peg with three lateral springs k_X, k_y, and k_Z and three angular springs k_{mx}, k_{my}, k_{mz}. Both the operator-



smart_hand#2 to smsr_taskboard min distance = 66.1070 mm

Global min distance = 66.1070 mm

Figure 13. Virtual Reality Proximity Display. (Black bars in the upper right hand corner show the distance of the two fingerlips relative to the vertical surface of the block, in a total range of 100 mm. As the text indicates, the actual value of the shown distance is 66.4 mm.)

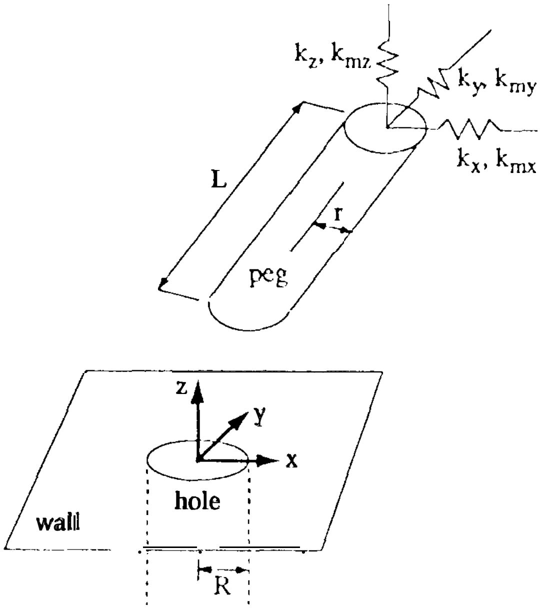


Figure 14. Geometry of a Simulated Peg-in-Hole Task with Lateral and Angular Springs at the Compliance Center

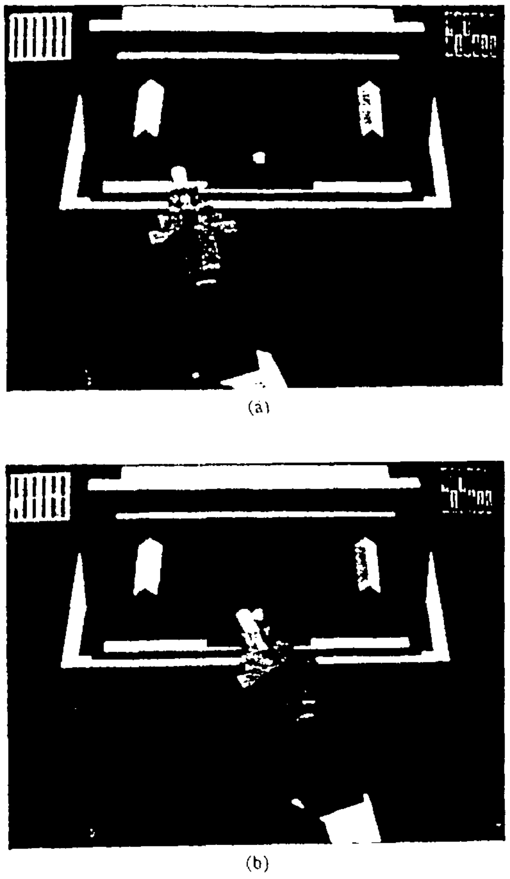


Figure 15. Force-Reflecting Teleoperation Training Displays; (a) before contact, (b) after contact with the wall

commanded and the actual positions of the peg are described by the position of the compliance center. For a given operator-commanded peg position, the actual peg position after compliant accommodation can be different, depending upon the current state of the peg of whether the peg is currently in the hole or not. For the peg-not-in-hole state, two conditions are considered: no-touch or peg-on-wall. For the pcg-itl-hole state, four conditions arc considered: no-touch, pc~,-side one-point contact, peg-tip one-point contact, m two point contact.

Figure 15 shows a force-reflecting virtual reality training display for a pcg-in-hole task. Contact forces and torques arc computed and reflected back to a force-reflecting hand controller in real time. They arc also displayed on the upper left corner of the display screen.

6. CONCLIJS1ONS

Some of the described sensor data fusion display and control algorithms underwent laboratory tests and evaluation. For instance, the system dcscribcd in Figures 7 through 9, was tested with astronauts at the Johnson Space Center (JSC) using the hardware simulation of the Space Shuttle Remote Manipulator Systcm (RMS). The tests clearly demonstrated the utility of the fused sensor data display systcm. The results have shown that, on the average, positioning and alignment accuracy improved by a factor of three, and task completion time was rcduccd by a factor of two [3].

Acknowledgment

This work was carried out at the Jet Propulsion Laboratory, California Institute of Technology under contract by NASA. The contribution of Dr. G. Paine of JPL to some of the event-driven display work is gratefully acknowledged.

References

[1] W. S. Kim and A. K. Bcjczy, "Graphics Displays for Opcιator Aid in Tclcmanipulation," *Proceedings of IEEE Int'l Conf. on Systems, Man and Cybernetics,* Charlottesville, VA, Oct. 13-16, 1991.

[2] W. S Kim and A.K.Bcjczy, "Dcmonstration of a High-Fidclity Predictive/I'review Display Technique for Telcιobotic Servicing in Space, *"IEEE Transactions on Robotics and Automation,*, Special Issuc on Space Telcιobotics, Oct. 1993.

[3] A. K Bcjczy, J. W. Brown and J. L. Lewis, "Evaluation of Smart Sensor Displays for Multidimensional Precision Control of Space Shuttle Remote Manipulator," *Proceedings of 16th Annual Conference on Manual Control,* Massachusetts Institute of Technology, Cambridge, MA